

# 1. Describe in detail what the components of the invention are and how the invention works

## INVENTION

As long as computer generated graphics are represented by a finite number of screen pixels, there will be some visual anomalies called the jaggies or staircasing, a phenomenon known as aliasing. The application of techniques that reduce aliasing is called anti-aliasing, of which full-scene anti-aliasing via supersampling is one of these techniques. Other anti-aliasing methods include sub-pixel computations, edge blending, and color accumulation.

Supersampling is a simple approach to full-scene anti-aliasing in which the original scene is rendered at a higher resolution and then filtered down to the original screen resolution. This in effect raises the Nyquist limit, which simply shifts the aliasing effect up to a higher spatial frequency. Though the technique does not eliminate aliasing completely, the method is simple and widely used by many 3D graphics accelerators today. However, there are some performance drawbacks when this technique is employed, namely the extra processing, and memory storage and bandwidth required to render the image at  $k$  times the original resolution and later filtered down. Supersampling two times in each of  $x$  and  $y$  directions ( $k=4$ ) results in four times of processing, storage, and bandwidth. This patent describes an efficient implementation of supersampling without incurring extra memory storage and bandwidth by using a tile-based rendering architecture in conjunction with a unified graphics cache.

In a tile-based rendering architecture with a unified graphics cache model, the color and depth values for pixels inside each tile are stored in the "Graphics Color/Z" partition of the unified cache (Figure 1). The tile size is determined based on the color and depth formats and the size of the unified graphics cache. For instance, a 64KB unified graphics cache can accommodate a tile size of 128x64 pixels, each pixel consisted of 32bit color and depth values. Besides color and depth data, the unified graphics cache also stores the texture data.

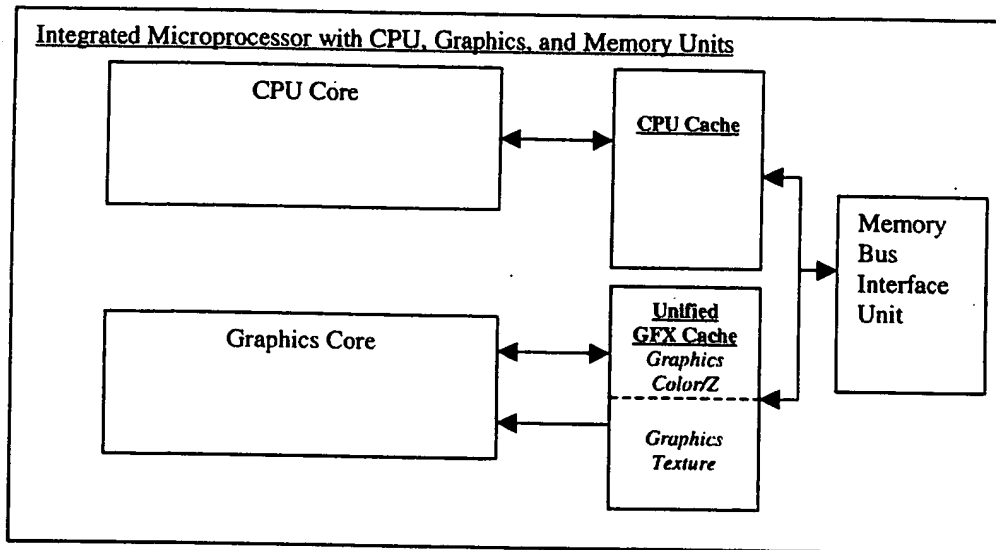
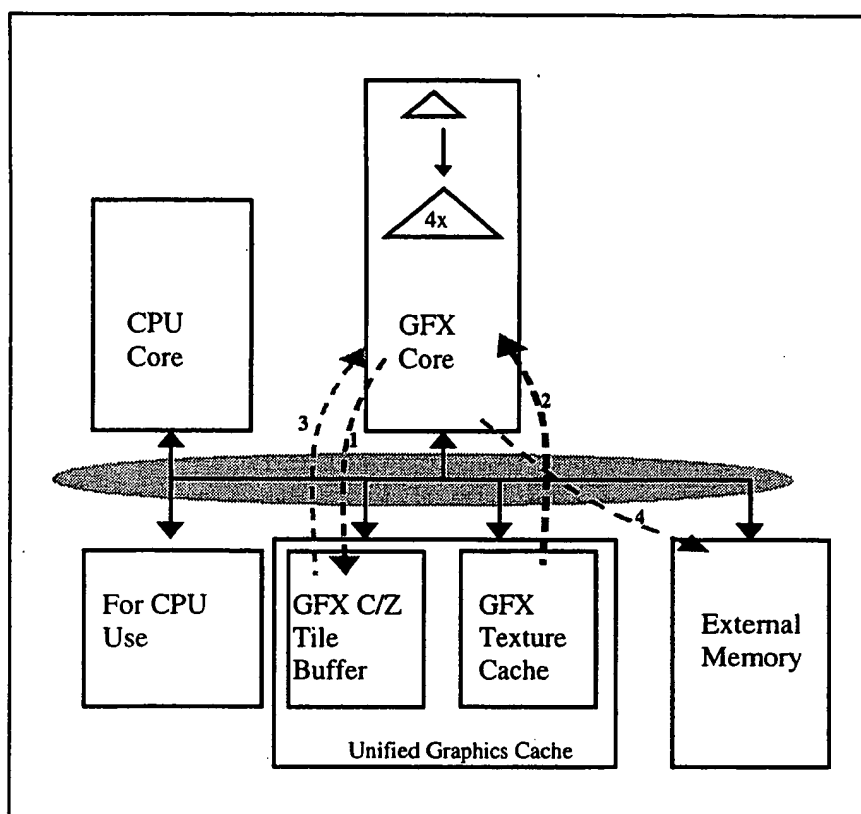


Figure 1: A L2 Sharing Model in an integrated microprocessor using tile-based 3D rendering architecture

Under a tile-based rendering architecture, each tile is rendered one at a time. The "Graphics Color/Z" cache is large enough to fulfill intermediate color and Z data accesses for all triangles that fall inside the tile. The color and Z data are written back to the external frame buffer after the last triangle in the tile finishes rendering. All pixels in the tile are then considered complete, and will not be rendered again. The purpose of this invention is to exploit the benefit of a tile-based architecture in conjunction with the micro-architectural features of a unified graphics cache to perform supersampling efficiently.

The patent covers efficient implementation of supersampling by eliminating the extra memory storage and bandwidth requirements using the unified graphics cache model. Figure 2 below describes the data flow when supersampling is enabled. For simplicity, a value of  $k=4$  is assumed by supersampling 2x in both X and Y directions. The technique is equally applicable to any other  $k$  values. Additionally, the "physical" tile size is assumed to be 128x64, and polygons are software-binned into a "virtual" tile size of 64x32.

EXHIBIT A



**Figure 2: Data Flow Diagram for Efficient Supersampling Using Tile-Based Rendering with a Unified Graphics Cache**

1. Polygons, already binned into tiles, are received in their original forms into the graphics core, but are internally amplified 4x of its original size. This is achieved through the viewport transformation supported by the graphics setup engine.
2. The enlarged polygons are tile-based rendered into the "GFX C/Z Tile Buffer" in the unified graphics cache (red line #1). Texture data for polygons being rendered are accessed from the "GFX Texture Cache" if hits occur to the cache (pink line #2).
3. After the last triangle in the tile finishes rendering, the "GFX C/Z Tile Buffer" contains the complete image of the tile that is 4x of its original tile size.
4. A stretch BLT is performed to down sample the image from the virtual tile down to the physical tile size. This is accomplished by rendering a rectangle (made up of two polygons) of the size equal to that of the physical tile size. The supersampled image in the virtual tile (still stored in the "GFX C/Z Tile Buffer") is considered the source of the stretch BLT, while the destination is allocated in the external memory. Internally, the micro-architecture treats the source of the stretch BLT as a texture map for the destination rectangle (blue line #3). The "GFX Texture Cache" is kept undisturbed to maintain good utilization of the texture data across tiles.
5. While the stretch BLT is occurring and its results being written out to the "External Memory" (green line #4), rendering of the next tile can begin in the pipelined engine.

## ADVANTAGES

Using a tile-based rendering and a unified graphics cache architecture, efficient anti-aliased (via supersampling) images can be created without increasing the external memory storage and bandwidth requirements. This is achieved through the use of the unified graphics cache that provides a temporary storage for the supersampled image to be later filtered down (through stretch BLT). In this way, only the final image of the original size needs to be written out and stored in the external frame buffer. On the contrary, a non-tiled based rendering engine must first render the entire supersampled image to a memory surface of the size of  $k$  times of the original screen resolution before the downsampling can occur. Typically, this memory is too large to be implemented on-chip which leads to an increase in the memory storage and bandwidth requirements.

This invention is valuable to an integrated microprocessor targeting the Value PC segment where cost is a primary concern. Because of the cost constraint, the memory subsystem bandwidth is often limited due to the number of memory channels available. By avoiding extra memory bandwidth and storage requirements, an integrated microprocessor can take advantages of the unified graphics cache and the tile-based rendering architecture to render images with full-scene antialiasing feature enabled with a minimal performance penalty.